



Dell | Small Business

Is your office
for here or to go?

Excel's Array Formulas

October 1, 2003
By Helen Bradley

Have you ever sat in front of your monitor pulling your hair out trying to identify duplicate entries in a list? If so, you should learn about Microsoft Excel's array formulas. In fact, you can use array formulas to perform calculations that are otherwise impossible in Excel, and you can enhance the power of some of the program's existing functions.

In Excel, an array is a block of adjacent cells that are treated as a group. To see this, highlight a block of cells, type a number, and then press Ctrl-Shift-Enter. You'll see the number in all the highlighted cells.

With array formulas, the array indicates that the formulas act on sets of values rather than a single value. Each set of values (known as an argument) must have the same number of rows and columns. You enter array formulas the same way as normal formulas except that, as above, you press Ctrl-Shift-Enter.

Though you can use array formulas to return either a single result or multiple results, we will keep things simple by focusing on the former type. Here's an array formula that calculates the total stock value from a list containing numbers of items in stock (numberInStock) and their prices (price):

```
=sum(price*numberInStock)
```

To enter this as an array formula, type it and press Ctrl-Shift-Enter. (If you press only Enter by mistake, simply click on the cell, press F2 and then press Ctrl-Shift-Enter.) If you select the cell, you'll see that the formula is enclosed in a set of curly brackets, which indicates it is an array formula.

The formula takes each value in the range price and multiplies it by the corresponding value in the range numberInStock. The results are then added, and a single value is returned. (Coincidentally, this formula performs the same task as Excel's Sumproduct function.) As you can with regular formulas, you can use range names (as we have), or actual cell references in array formulas.

Extending SumIF

Excel's SumIF function sums a series of numbers if a certain condition is true. So if you have a series of last names in column B (with the range named salesperson) and sales in column C (sales), the following formula sums the sales for the salesperson Smith:

```
=SUMIF(salesperson,"Smith",sales)
```

You can write this as an array formula:

```
=SUM(IF(salesperson="Smith",sales))
```

Extending this array formula to calculate multiple conditions is easier than attempting it with the SumIF function. So if there are month names in column A (month), the following array formula calculates the sales for Smith in January:

```
=SUM(IF((salesperson="Smith")*(month="Jan"),sales))
```

The formula works by comparing each value in the salesperson range with the word Smith and returns True (1) if there is a match and False (0) if not. It also compares each value in month with the word Jan and returns 1 for a match and 0 otherwise. When the results are multiplied, only entries where the salesperson is Smith and the month is January will return a True result and be passed to the Sum function to be added. (Note that any nonzero number returns a True result.)

Performing a calculation using OR criteria is similar. In this case you add the condition results instead of multiplying them. The following example calculates the total sales for January or February:

```
=SUM(IF((month="Jan")+(month="Feb"),sales))
```

Any of these examples can be easily adapted to your worksheets. Simply replace our range names with your own and type the words to match between the quotation marks. If you're using OR criteria, you can look up two items in the same range (as above) or two items from different ranges.

Check for Unique Data

The handy array formula below checks to see whether the contents of a column of numbers or text (dataRange) contains duplicate entries:

```
=MAX(COUNTIF(dataRange, dataRange))
```

It uses the CountIf function with the same range passed to it for both arguments (range and criteria). This forces Excel to compare every value in the range dataRange with every other value in the range to determine whether the values are the same. Each value will (of course) be equal to itself, but you need to know whether it is also equal to any other value (indicating duplicates in the range). Countif counts the number of matches for each comparison and passes the results to Max, which returns the maximum number of matches. If the result is 1, each number matches only itself; there are no duplicates. If the result is larger than 1, there are duplicates.

You can combine this formula with an IF function to give a textual explanation of the results:

```
=IF(MAX(COUNTIF(dataRange, dataRange))=1,"No Duplicates", "Duplicates")
```

When you're working with array formulas, start with a small amount of data on a fresh worksheet. This lets you test a formula to ensure its accuracy. You should also use named ranges in your formula instead of cell references, so you can copy the formulas to other worksheets that use the same named ranges.

Copyright (c) 2003 Ziff Davis Media Inc. All Rights Reserved.